WHAT IS CLAIMED IS:

1. A chip to chip interface comprising:

a driver configured to receive a data signal and provide an output signal at a first level in response to receiving an odd number of consecutive logic highs in the data signal, at a second level in response to receiving an odd number of consecutive logic lows in the data signal, at a third level in response to receiving an even number of consecutive logic highs in the data signal and at a fourth level in response to receiving an even number of consecutive logic lows in the data signal.

2. The chip to chip interface of claim 1, comprising a receiver configured to receive the output signal and provide a clock signal and data from the data signal.

3. The chip to chip interface of claim 2, wherein the receiver is configured to provide the data as even data and odd data.

4. The chip to chip interface of claim 2, wherein the clock signal is a 500 MHz clock signal and the data signal is a 1 GHz double data rate data signal.

5. The chip to chip interface of claim 1, comprising a receiver configured to compare the output signal to four reference levels to provide data from the data signal.

6. The chip to chip interface of claim 1, comprising a receiver configured to compare the output signal to four reference levels to provide a clock signal.

7. The chip to chip interface of claim 1, wherein the fourth level is less than the second level that is less than the first level that is less than the third level.

8. The chip to chip interface of claim 1, comprising a receiver configured to toggle a data clock signal in response to each transition in the output signal.

9. The chip to chip interface of claim 8, wherein the data clock signal latches data from the data signal.

10. The chip to chip interface of claim 1, wherein the data signal is a double data rate data signal.

11. A chip to chip interface comprising:

a driver configured to encode a data signal and a clock signal into an encoded signal by driving the encoded signal to a first reference level at a first edge of the clock signal in response to a logic low to a logic high transition in the data signal, driving the encoded signal to a second reference level at a second edge of the clock signal in response to a logic high to a logic low transition in the data signal, driving the encoded signal to a third reference level at a third edge of the clock signal in response to no change in the data signal if the encoded signal was at the first reference level, and driving the encoded signal to a fourth reference level at a fourth edge of the clock signal in response to no change in the data signal if the encoded signal was at the second reference level; and

a receiver configured to receive the encoded signal.

12. The chip to chip interface of claim 11, wherein the driver is configured to drive the encoded signal to the first reference level at a fifth edge of the clock signal in response to no change in the data signal if the encoded signal was at the third reference level.

13. The chip to chip interface of claim 11, wherein the driver is configured to drive the encoded signal to the second reference level at a fifth edge of the clock signal in response to no change in the data signal if the encoded signal was at the fourth reference level.

14. The chip to chip interface of claim 11, wherein the second reference level is less than the first reference level, and the third reference level is greater than the first reference level, and the fourth reference level is less than the second reference level.

15. A memory interface comprising:

a memory controller comprising a driver configured to receive a clock signal and a data signal and provide an output signal that changes at each edge of the clock signal to one of four levels based on the data signal; and

a memory comprising a receiver configured to receive the output signal and compare the output signal to reference levels to recreate the clock signal and obtain data from the data signal in the receiver.

16. The memory interface of claim 15, wherein the one of four levels chosen for the output signal is based on a current data bit and a previous state of the driver.

17. The memory interface of claim 15, wherein the driver comprises two p-channel drivers and two n-channel drivers configured to supply the four levels.

18. The memory interface of claim 15, wherein the receiver comprises comparators configured to provide comparator outputs to indicate the one of four levels of the output signal.

19. The memory interface of claim 18, wherein the receiver comprises four comparators and each comparator is configured to compare the output signal to one of four reference signals.

20. The memory interface of claim 15, wherein the receiver comprises a clock decoder configured to toggle the clock signal at each change in the output signal to recreate the clock signal in the receiver.

21. The memory interface of claim 15, wherein the receiver is configured to separate the data obtained from the output signal into even data and odd data.

22. The memory interface of claim 15, wherein the memory is a dynamic random access memory.

23. The memory interface of claim 15, wherein the memory is a double data rate dynamic random access memory.

24. A chip to chip interface comprising:

means for providing an encoded signal comprising a clock signal and a double data rate data signal on a single signal path; and

means for decoding the encoded signal to recreate the clock signal and obtain data from the double data rate data signal.

25. The chip to chip interface of claim 24, wherein the means for providing an encoded signal comprises means for providing the encoded signal at a first level in response to a change in the double data rate data signal from a low level to a high level, at a second level in response to a change in the double data rate data signal from a high level to a low level, at a third reference level in response to consecutive high levels in the double data rate data signal, and at a fourth reference level in response to consecutive low levels in the double data rate data signal.

26. The chip to chip interface of claim 24, wherein the means for decoding the encoded signal comprises means for toggling a data clock signal at each transition of the encoded signal.

27. A method of passing data and a clock signal between chips comprising:

driving a signal to a first level at a first edge of the clock signal to indicate a change in a data stream from a logic low to a logic high;

driving the signal to a second level at a second edge of the clock signal to indicate a change in the data stream from a logic high to a logic low;

driving the signal to a third reference level at a third edge of the clock signal to indicate consecutive logic highs in the data stream; and

driving the signal to a fourth reference level at a fourth edge of the clock signal to indicate consecutive logic lows in the data stream.

28. The method of claim 27, comprising:

driving the signal alternately between the first level and the third level at consecutive edges of the clock signal to indicate more than two consecutive logic highs in the data stream; and

driving the signal alternately between the second level and the fourth level at consecutive edges of the clock signal to indicate more than two consecutive logic lows in the data stream.

29. The method of claim 27, comprising:

comparing the signal to a plurality of reference levels to determine the data stream.

30. The method of claim 29, wherein the plurality of reference levels comprises four reference levels.

31. The method of claim 29, comprising:

toggling a data clock signal at each transition of the signal.